SIGNALS AND SYSTEMS MATLAB ASSIGNMENT

By : Jash Shah

2018A8PS0507P

T – 5 [Shishir Maheshwari]

Q1 a) Generate a single-frequency cosine signal of amplitude = 2 and duration = 2 seconds (i.e., from 0 to 2 secs). The frequency (in Hz) is 507Hz. Assume that the sampling frequency = 20kHz. Make sure a single cycle corresponds to a fundamental period.

Q1b) Now, generate a new signal which is same as the signal in Q1a) but delayed by 100 msec.

Plot these signals as a function of time (up to 2 cycles). Note: Show results as subplots of 2x1.

Solution -

-> **Matlab code**

This is question 1

1.a The question requires the creation of cosine wave with the given credentials:-

Sampling frequency = 20kHz i.e. number of time samples to be used are 20,000

```
sam_freq = 2e004;                    % represented in scientific notation
t = linspace(0,2,2*sam_freq);
```

Frequency is last three digits of my Id. Hence. freq = 507 Hz (My Id is 0507)

```
f = 507;
```

Amplitude give is 2 units

```
sig_1a = 2*cos(2*pi*f*t);
```

1.b In this part, I will need to delay the time by100msec.

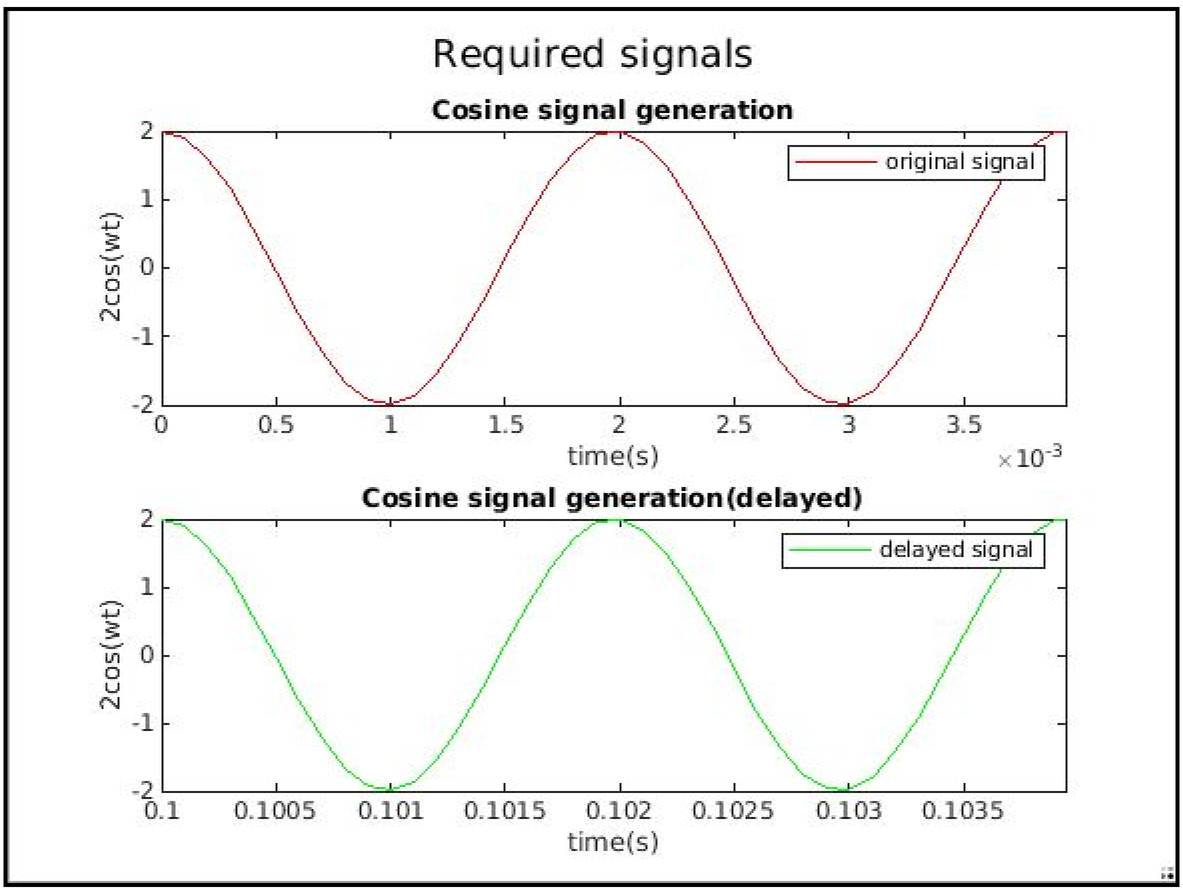
```
error = 0.1;
correction = error*(ones(1,2*sam_freq));
new_t = t + correction;
sig_1b = 2*cos(2*pi*f*t);
```

Plotting the signal till 2 cycles only

```
subplot(2,1,1),plot(t,sig_1a,'r'), legend('original signal');
title('Cosine signal generation');
xlabel('time(s)');
```

```
ylabel('2cos(wt)');
axis([0 2/f -2 2]);


subplot(2,1,2),plot(new_t,sig_1b,'g'), legend('delayed signal');
title('Cosine signal generation(delayed)');
xlabel('time(s)');
ylabel('2cos(wt)');axis([0.1 ((2/f)+0.1) -2 2]);


suptitle('Required signals');
```

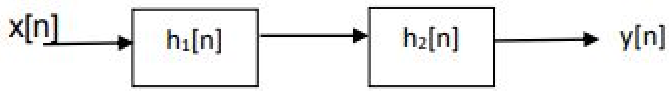
-> **Output Figure**



-

---------------X---------------------------X---------------------------X---------------------------X

Q2) Consider the LTI systems shown below:



x[n] = u[n - 1] – u[n - 6]

h1[n] = 1 for $(-7 \le n \le 4)$ and 0 otherwise.

h2[n] = u[n - 1] - u[n - 2]

Plot x[n], h1[n], x[n]*h1[n], and y[n] as subplots of 2x2.

<u>Solution</u> –

-> **Matlab code**

1) <u>Unit step function declaration :</u>

This is function file to define unit step impulse.

```
function[u] = u(n)
    len = length(n);
    u = zeros(1,len);
    for i = 1:1:len
        if n(1,i)>=0
            u(1,i) = 1;
        end
    end
end
```

2) <u>h1 function declaration according to the question:</u>

Function for calculating h1[n]

My Id is 0507, so a = 7.

```
function [h1] = h1(n)
    len = length(n);
    h1 = zeros(1,len);
    for i = 1:1:len
        if (n(1,i)>=-7 & n(1,i)<=4)
            h1(1,i) = 1;
        end
```

```
    end
end
```

3) Main body :

Q2 OF ASSIGNMENT.

Defining the signals

```
n = -20:1:20;



x = u(n-1) - u(n-6);
h2 = u(n-1) - u(n-2);

sig_2 = h1(n);



x1 = conv(x,h1(n));
y = conv(x1,h2);
```

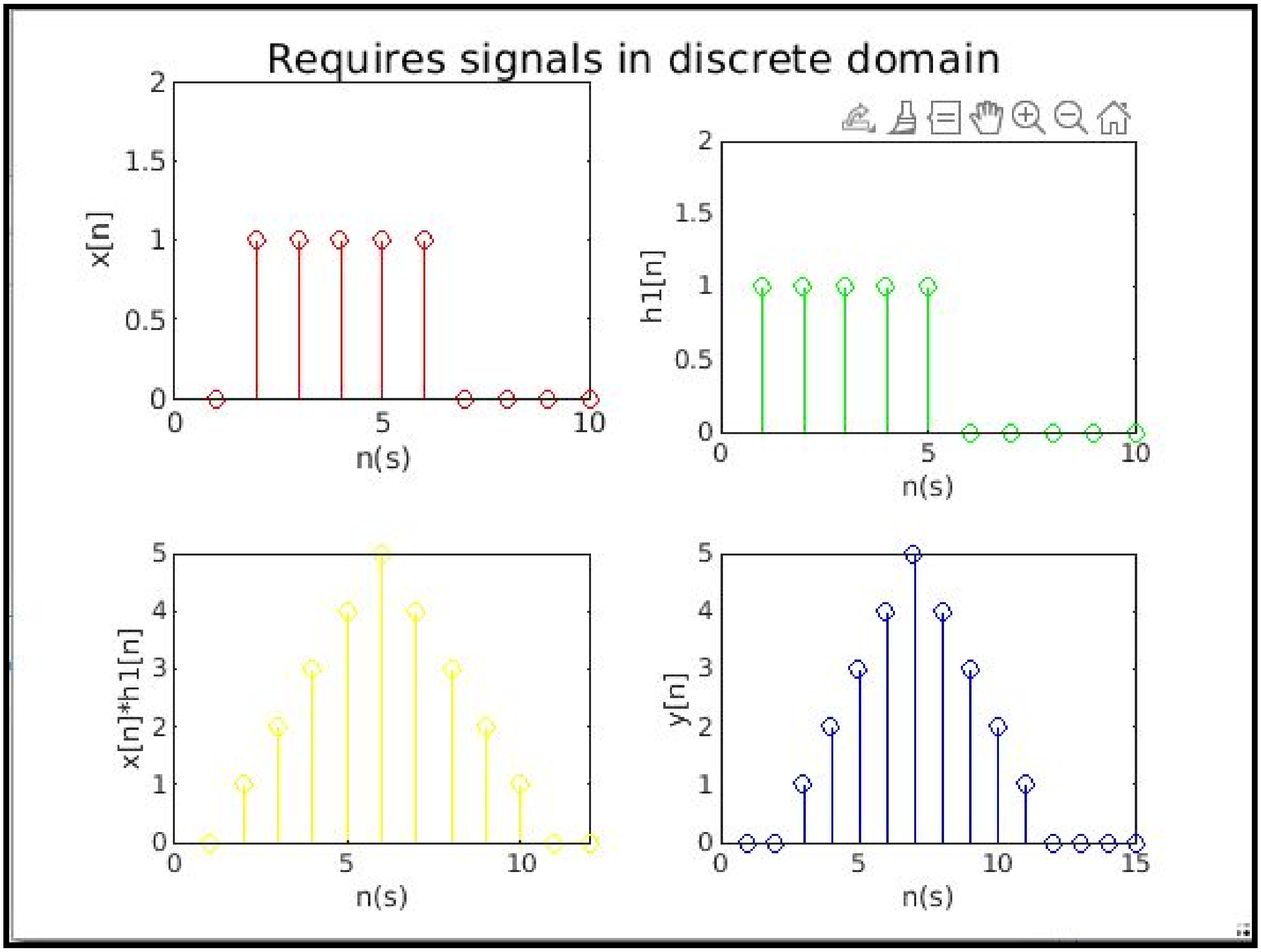
Plotting the signals now.

```
subplot(2,2,1), stem(x,'r'), ylabel('x[n]'), xlabel('n(s)'), axis([0 10 0 2]);

subplot(2,2,2), stem(sig_2,'g'), ylabel('h1[n]'), xlabel('n(s)'), axis([0 10 0
2]); % The points other than that shown in the plot have zero amplitude.

subplot(2,2,3), stem(x1,'y'), ylabel('x[n]*h1[n]'), xlabel('n(s)'), axis([0 12 0
5]);

subplot(2,2,4), stem(y,'b'), ylabel('y[n]'), xlabel('n(s)'),axis([0 15 0 5]);

suptitle('Requires signals in discrete domain'); % To add a common or super title
to all the plots.
```

-> **Comments**

- The points that are not shown in the plot have zero amplitude.
- The axis was manually set by me so as to best fit the stem plot.

-> **Output Figure**



------------X---------------------------X--------------------------X------------------------------X

Q3 The system function of a causal LTI system is given by:

$$H(z) = (5z^2 + 1.5z + 7)/(7z^2 + 1.5z + 5)$$

Sketch the pole-zero diagram and comment on its stability. Additionally, display the locations of poles and zeros.

Solution –

My Id being 0507,

a = 5 b = 1.5(given parameter if b=0) c = 7

-> **MATLAB code**

THIS IS Q3 OF ASSIGNMENT.

```
sys = tf([5,1.5,7],[7,1.5,5]);

z = zero(sys);
p = pole(sys);
```

Now plotting the pole-zero diagram.

```
len_p = length(p);

len_z = length(z);



for i = 1:1:len_p
    plot(real(p(i)),imag(p(i)),'bX')

    textString1 = sprintf('(%d, %d)', real(p(i)), imag(p(i)));

    text(real(p(i))-0.03, imag(p(i))+0.1, textString1, 'FontSize', 7);

    hold on

end



for j = 1:1:len_z
```

```
    plot(real(z(j)),imag(z(j)),'rO')

    textString2 = sprintf('(%d, %d)', real(z(j)), imag(z(j)));

    text(real(z(j))-0.03, imag(z(j))+0.1, textString2, 'FontSize', 7);

end

xlabel('Real axis')

ylabel('Imaginary axis')

grid on
title('Pole - Zero diagram')

axis([-0.2 0.2 -1.5 1.5])
```

Marking the co-ordinate axis for better view of stability.

```
x_abscissa = [-1 1]

y_abscissa = [0 0]



plot(x_abscissa,y_abscissa,'color', 'black')

x_ord = [0 0]
y_ord = [x-2 2]

plot(x_ord,y_ord,'color', 'black')
```

Creating the circles now

```
% for unit circle:
a = -pi:0.001:pi;
x_u_cir = cos(a);
y_u_cir = sin(a);


plot(x_u_cir,y_u_cir,'k--')
hold off;
```

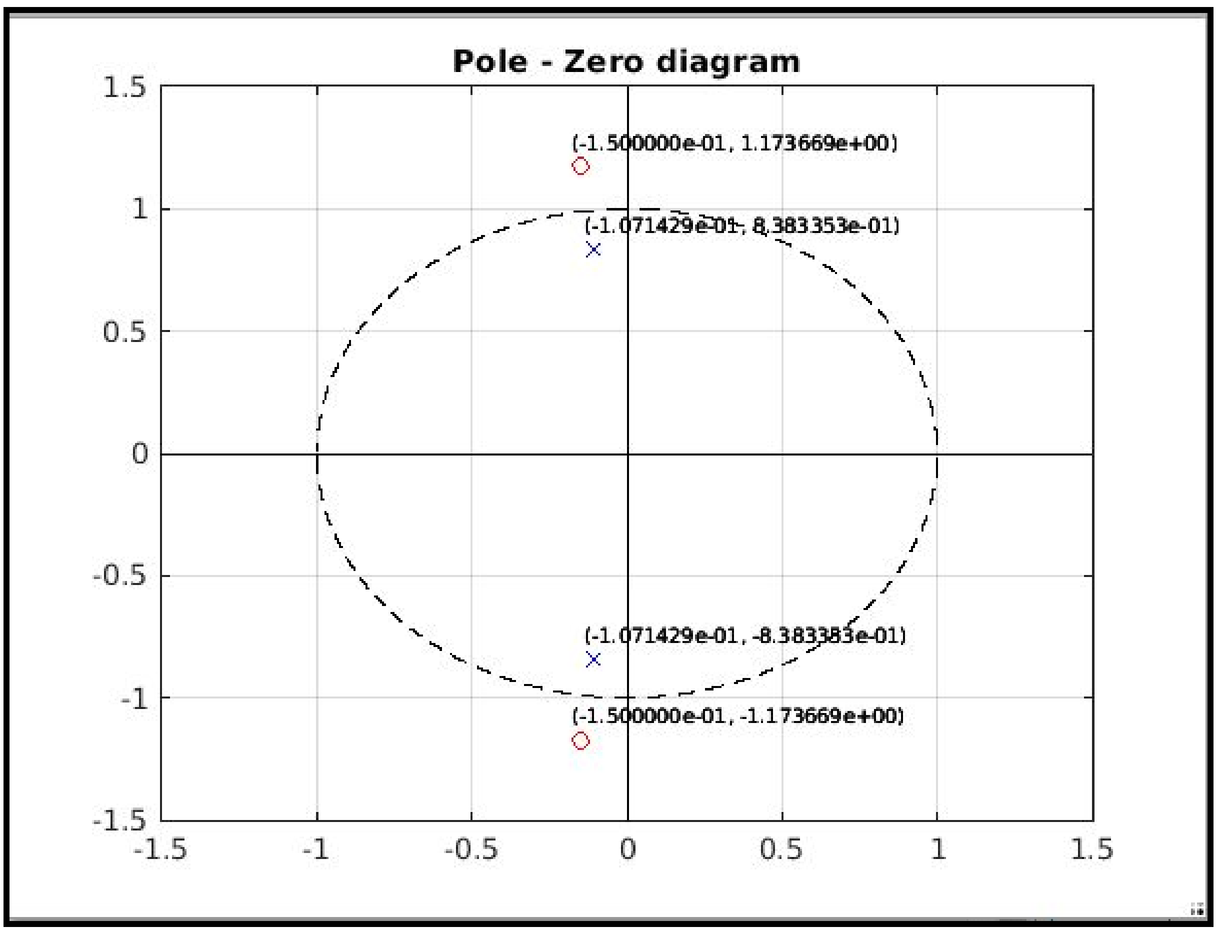
-> **Observations**

- We can clearly see that the transfer function has a relative degree of 0. Hence, the function is causal with zero feedback or instantaneous transfer between input and output.

- For a causal system, the mode(z) for poles must be greater than something. In our case we don't have a choice and thus, it's greater than the created pole circle.

- We also note that in doing so, the ROC crosses mode(z) = 1 thus making it a stable system.

-> **Output Figure**



1) Unit circle is made

-> **Conclusions and comments**

- For the system to be stable and causal, mode(z) must be greater than the mode(poles).i.e. the system must be completely right-handed or on the positive axis of time. Which is proved by the hand calculations above.

- Also, the poles marked on the left side of the real axis, this means that the system is bounded . It has no exponentially increasing component.

------------X---------------------------X--------------------------X------------------------------X

Q4) An input signal is a combination of two cosine signals. The first cosine signal has a frequency equal to 507 Hz and the second cosine signal contains a frequency of 11507Hz. Note that the individual cosine signal generation is the same as in Q1a; i.e., amplitude = 2, duration = 2 seconds and sampling frequency = 20kHz). Sketch the magnitude spectrum of the input signal. Make sure your x-axis is in frequency (in Hz), ranging between 0 to 10kHz. Comment on your observations.

Solution –

My Id – 0507. So,

freq1 = 507Hz and freq2= 11507Hz

-> **MATLAB code**

This is Q4 of the assignment.

Defining the signals

```
fsampling = 20000;

t = linspace(0,2,2*20000);
amp = 2;



freq_1 = 507;
freq_2 = 11507;



sig_1 = amp*cos(2*pi*freq_1*t);
sig_2 = amp*cos(2*pi*freq_2*t);

inp_sig = sig_1 + sig_2;



subplot(3,1,1),plot(t,sig_1);
title('Signal 1 i.e. freq = 507Hz'), axis([0 2/507 -2 2]);

subplot(3,1,2),plot(t,sig_2);

title('Signal 2 i.e. freq = 11507Hz'), axis([0 2/11507 -2 2]);

subplot(3,1,3),plot(t,inp_sig);

title('Combined Signal'), axis([0 2/507 -4 4]);
```

Now doing the fourier transform and creating the required Magnitude function of frequency.

```
y = fft(inp_sig);

len_y = length(y);

x = linspace(0,10000,len_y);
```

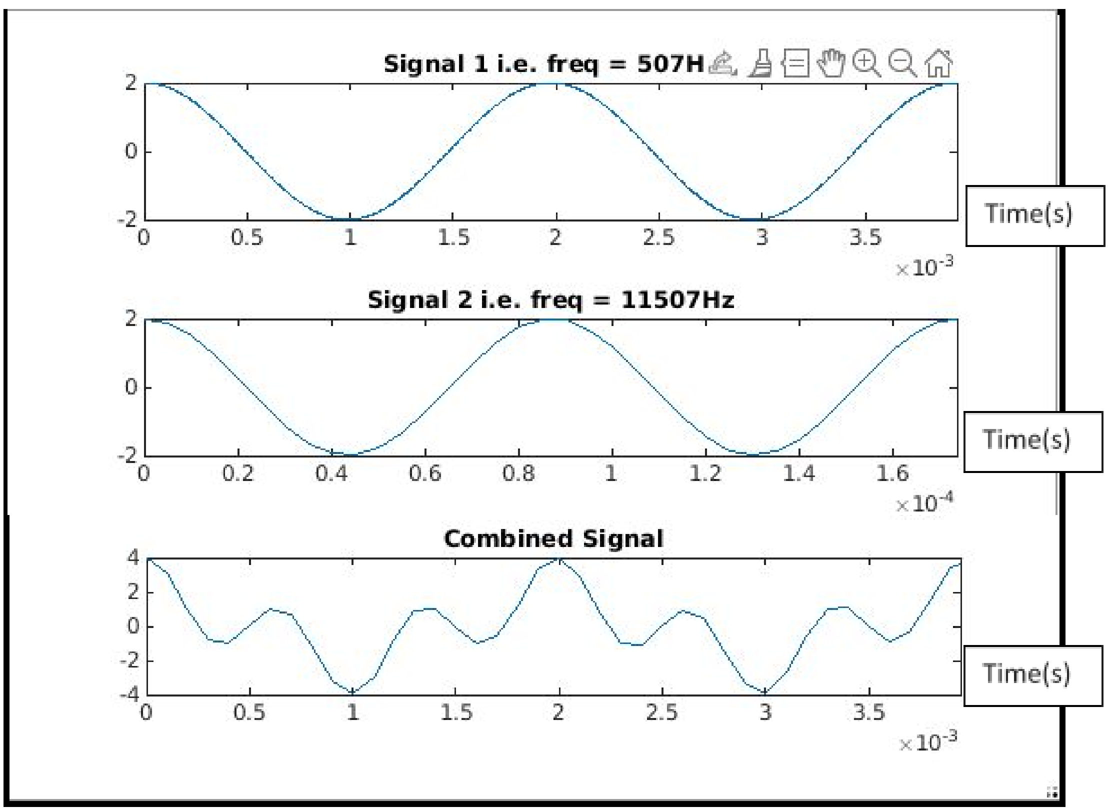
Plotting the magnitude spectrum.

```
hold off

plot(x,2*abs(y/20000));

xlabel('Frequency (Hz)');
```

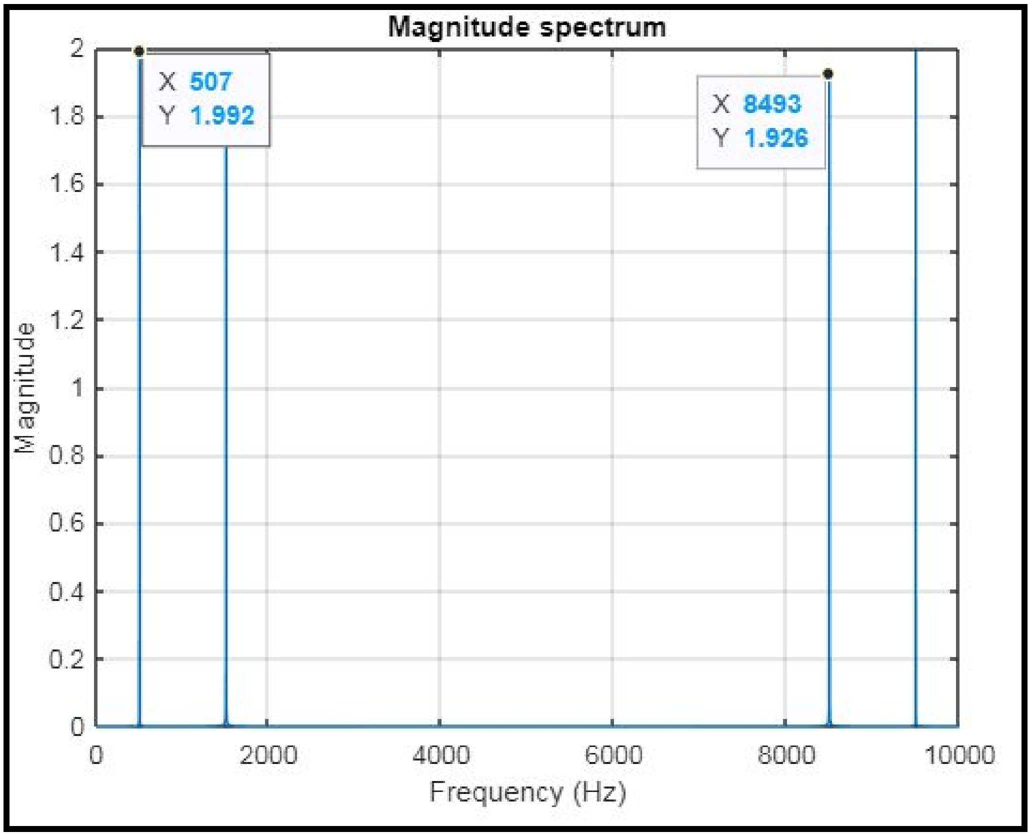
```
ylabel('Magnitude');

title('Magnitude spectrum');

grid;
```

-> **Output Figure**

1) The three signals plotted with respect to time in seconds



2) Magnitude Spectrum of the combined signal



-> **Observations**

- All the 4 peaks have a magnitude nearing to 2 i.e. the magnitude or amplitude of the original cosine functions.

- The graph has 4 peaks and I will justify them one by one:

    - 1st peak – It is at frequency of 507Hz owing to sig_1 or the cosine corresponding to 507Hz frequency in input signal.

    - 2nd peak – It is basically formed to represent (-11507) Hz frequency. {2cos x = e^jx + e^-jx}. So, it's the -jx part and because of change in axis, the frequency is shown over here.

- 3rd peak – It shows 11507 Hz frequency. Now because of constraints in axis and due to time period complications in FFT, it is shown as (20000 – 11507) Hz .

- 4th peak - It is owing to the -507 Hz frequency and is formed in symmetry, just like peak 2 is formed.